Concevoir des pages accessibles à tous : introduction à l’accessibilité web

C

Concevoir des pages accessibles à tous : introduction à l'accessibilité web

1. **Introductions aux balises ARIA**
2. **Rôles ARIA**
3. **Attributs ARIA**
4. **Navigation au clavier & lecteurs d'écran avec tabindex & outline**
5. **SEO & Accessibilité**
6. **Exercice**
7. **Ressources**

Introduction aux Balises ARIA (Accessible Rich Internet Applications)

ARIA est un ensemble d'attributs permettant d'améliorer l'accessibilité des sites web pour les technologies d'assistance (lecteurs d'écran, logiciels de reconnaissance vocale, etc.).

Rôle des attributs ARIA en HTML

Les rôles ARIA définissent la fonction d'un élément dans la page.

```
<input type="button" role="button" value="Acheter">
```

Attributs ARIA en HTML

Les attributs ARIA permettent d'améliorer la compréhension des éléments interactifs.

```
<input arial-label="Nom" placeholder="Votre nom">
```

Principaux rôles ARIA

Rôle ARIA	Utilisation
role="button"	Indique un bouton interactif.
role="alert"	Message d’alerte lu immédiatement.
role="navigation"	Désigne une barre de navigation.
role="dialog"	Définit une boîte de dialogue.
role="progressbar"	Définit une barre de progression.
role=”form”	Indique une section qui a le rôle d’un formulaire (<div role=”form”)
role=”search”	Indique un formulaire de recherche (<form role=”search”>)
Tout voir : https://developer.mozilla.org/en-US/docs/Web/Accessibility/ARIA/Reference/Roles	

- Évitez d’utiliser ARIA si HTML natif suffit (<button> est déjà accessible, pas besoin de role="button").
- Testez votre site avec un lecteur d’écran (ex: NVDA, VoiceOver, JAWS).

Principaux attributs ARIA

Rôle ARIA	Utilisation
aria-label	Donne un nom accessible à un élément.
aria-hidden	Cache un élément aux lecteurs d’écran (true ou false).
aria-live	Indique aux lecteurs d’écran qu’un élément va être mis à jour dynamiquement.
aria-expanded	Indique si un menu ou un bouton est ouvert (true ou false).
aria-describedby	Associe un élément à une description.
Tout voir : https://developer.mozilla.org/en-US/docs/Web/Accessibility/ARIA/Reference/Attributes	

Utilisez **aria-hidden="true"** sur les éléments décoratifs pour les cacher des lecteurs d’écran.

Navigation au clavier & lecteurs d'écran

Rendre son Site Navigable au Clavier

Tous les éléments interactifs doivent être accessibles avec la touche Tabulation (Tab).

```
<a href="#" tabindex="0">Lien accessible</a>
```

- **tabindex="0"** : Rend un élément focusable.
- **tabindex="-1"** : Retire un élément du focus clavier.

Bonnes pratiques

- **Utilisez tabindex=0** pour inclure un élément dans l'ordre de tabulation naturelle du contenu, mais rappelez-vous qu'un élément qui peut recevoir le focus par défaut est une option plus sûre qu'un contrôle personnalisé.
- **Utilisez tabindex=-1** pour donner le focus à un élément par programmation, tout en l'excluant de l'ordre de tabulation du contenu.
- **Évitez d'utiliser tabindex=1+.**

Navigation au clavier & lecteurs d'écran

Focus visuel

Ajoutez un style visible pour les éléments au focus :

```
<style>
  a:focus, button:focus {
    outline: 2px solid blue;
  }
</style>
```

- **Évitez outline: none;** car cela empêche les utilisateurs de voir où ils naviguent.

Compatibilité avec les lecteurs d’écran

Les lecteurs d’écran lisent les contenus HTML en fonction de la structure du document. Assurez-vous d’utiliser des balises sémantiques :

Balise HTML	Usage recommandé
<header>	En-tête de la page.
<nav>	Menu de navigation.
<main>	Contenu principal.
<article>	Article autonome.
<aside>	Contenu complémentaire.
<footer>	Pied de page.
<section>	Diviser une page en différentes sections logiques

Seo & accessibilité

L’**accessibilité web** est étroitement liée au **SEO** (référencement naturel).

Un site bien structuré et accessible améliore le classement sur les moteurs de recherche.

Élément	Usage recommandé
<title>	Décrit le contenu de la page (affiché dans les moteurs de recherche).
<meta name="description">	Brève description pour Google et les lecteurs d’écran.
<h1> à <h6>	Hiérarchie du contenu (titres bien ordonnés).
<img alt="">	Texte alternatif pour les images (utile pour SEO et accessibilité).
<a href="#" title="">	Donne un contexte supplémentaire aux liens.

Outils Accessibilité

Installez les outils suivants sur Chrome pour vous aider dans vos futurs projets :

- **Color Contrast** : https://chromewebstore.google.com/detail/color-contrast-checker-by/hfcoldjibhgbnlambdiecfneiegagkhi
- **BrowserStack Accessibility Toolkit :** https://chromewebstore.google.com/detail/fmkhjeeeojocenbconhndpiohohajokn?utm_source=item-share-cb
- **A11y quick check :** https://chromewebstore.google.com/detail/jlamgighkcjniljcdfpnhiemcakibepi?utm_source=item-share-cb
- **IBM Equal Access Accessibility Checker** : https://chromewebstore.google.com/detail/lkcagbfjnkomcinoddgooolagloogehp?utm_source=item-share-cb

Exercice

Vous allez corriger une page non accessible pour la rendre utilisable par tous les utilisateurs,

y compris ceux utilisant un clavier ou un lecteur d’écran.

https://www.afecdax.ovh/ressources/dev/html/exercices/accessibilite/exercice_accessibilite.html

Améliorations attendues :

✅ Ajouter des rôles ARIA (role, aria-label, aria-labelledby, aria-hidden).

✅ Corriger la navigation clavier (tabindex, aria-expanded, focus).

✅ Ajouter des textes alternatifs et des descriptions claires (alt, aria-describedby).

✅ Améliorer les interactions pour garantir une utilisation fluide et intuitive.

✅ Ajouter les balises sémantiques

Exercice (suite)

Partie 1 : Améliorations Accessibilité avec ARIA

- ✅ Ajoutez des rôles ARIA pour donner plus de contexte aux lecteurs d'écran.
- ✅ Utilisez aria-labelledby pour lier un titre à son contenu.
- ✅ Ajoutez aria-hidden="true" aux éléments purement décoratifs.

Partie 2 : Navigation Clavier & Interactions

- ✅ Ajoutez tabindex="0" aux éléments non focusables par défaut.
- ✅ Faire fonctionner le menu dépliant en JS et le rendre accessible.
- ✅ Ajoutez une gestion des focus visibles (outline en CSS) pour les utilisateurs clavier.

Partie 3 : SEO & Expérience Utilisateur

- ✅ Ajoutez des textes alternatifs (alt) aux images.
- ✅ Donnez des titres clairs aux liens (title).
- ✅ Assurez-vous que la hiérarchie des titres est logique (h1, h2, h3).

Ressources

Rôles ARIA
https://developer.mozilla.org/en-US/docs/Web/Accessibility/ARIA/Reference/Roles

Attributs ARIA
https://developer.mozilla.org/en-US/docs/Web/Accessibility/ARIA/Reference/Attributes

Application Lecteur NVDA
https://www.nvda.fr/index.php

Tabindex
https://developer.mozilla.org/fr/docs/Web/HTML/Global_attributes/tabindex

Outline
https://developer.mozilla.org/en-US/docs/Web/CSS/outline#accessibility

Balises sémantiques
https://developer.mozilla.org/fr/docs/Glossary/Semantics

Accede — Notice d'accessibilité HTML et CSS
https://www.accede-web.com/notices/html-et-css/

Exercice
https://www.afecdax.ovh/ressources/dev/html/exercices/accessibilite/exercice_accessibilite.html